

## LEGAL MODULE

### A WORK IN PROGRESS

## 1. TECHNICAL REQUIREMENTS

### ***1.1. Introduction***

This document is the technical response to the SWSS Foster Care Requirement for Legal. It will describe how the development team will implement the changes and additions to SWSS Childrens to effect the requirements. It is intended to explain to Policy Staff, who authored the requirements, how SWSS will behave once the solution is implemented, and is intended to explain to Project Staff how they might test the solution to verify its correctness.

This document is also to be used as a tool by the development team when coding the solution or maintaining it in the future. Thus this document is likely to be updated during the lifecycle of the SWSS project. Versions of this document will be maintained in PVCS, and the reader should be aware that multiple printed versions may exist.

### ***1.2. Module Description***

The purpose of the legal section is to keep a history of the petitions submitted to the Court, related hearings and orders resulting from those hearings. History will be displayed regardless of the log number; data is retrieved from the database based on the child\_id in swss\_case. In addition, future hearings can be recorded; a tickler reminding the worker is generated, a notice of this hearing to the foster parent/caretaker can be pre-filled and generated.

### ***1.3. Requirements***

#### **1.3.1. Process Description**

#### **1.3.2. Functional Requirements**

Legal automatically maintains some data for the user when certain actions are taken, especially in regards to adoption. The legal package on the database is the package containing the SQL code to retrieve and maintain the data.

#### **If the adoption is disrupted (order type 30, "Dismissal of Adoption Petition", is entered):**

- Database is maintained using legal.update\_misc\_legal\_info
- Save the adoptive parent(s) in the provider and provider\_person tables; provider\_type is "S" and provider\_id is provider\_id.nextval
- Find the current placement (prov\_end\_date will be null) in provider\_case\_person
  - Update the prov\_end\_date with the date of the order
  - Update prov\_end\_date\_entered with system date
  - Put the provider\_id and type from the previous step into the placement record.
  - Put "D" in adoption\_end\_reason

- Update cfc\_info
  - Set goal status 5 – Case Terminated-Client request
- Update the adoptive parent(s)’ information:
  - In case\_person, read previous\_relation for the adoptive parents
  - Update relation table for the adoptive parents:
    - Set relation\_type to the previous\_relation read from case\_person
  - Update case\_person table for the adoptive parents:
    - Set previous relation to “AP”
    - Set person\_status to “I” (Inactive status)
    - Set provider\_type to “S” (SWSS provider)
    - Set provider\_id to the one entered in provider\_case\_person
- Retrieve the cis\_cfc\_case\_no and cfc\_recipient\_id from swss\_conversion for this child and this log number
- Retrieve the child’s previous name from aka\_person\_name
- Update group\_person with the child’s info saved in swss\_conversion and aka\_person\_name.
- Update the current ad\_info record
  - Set the disruption\_date to the date of the order
- Update load\_case table
  - End the current primary worker
  - Add the CFC supervisor selected on the screen as the primary worker
  - End the current secondary worker if there is one.
- Update swss\_case
  - Change the primary\_worker\_id with the worker\_id of the CFC supervisor selected on the screen.
  - Set the case\_name back to the CFC name
  - Set the cis\_case\_no back to the CFC case number
  - Set the swss\_case\_state to 400 (Unregistered; this will allow the case to be re-companionated if so desired)
  - Set program\_code to “CFC”
  - Set latest\_legal\_status to the new legal status
- Modify the swss\_ini.ini file
  - Find the entry "Case Name"
  - Put the old CFC name back in there
  - Find the entry "Close\_Date"
  - Put the value in Frm.mskOrdDate as the value for "Close\_Date"
  - Find the entry "Close\_Code"
  - Put the value "2" (Adoption Disrupted) as the value for "Close\_Code"
  - Find the entry "Miscellaneous"
  - Put the value "Close Case" as the value for "Miscellaneous"
  - Find the entry "Source Process"
  - Put legal.exe as the value for "Source Process"
- Print the FIA5S after the above swss\_ini.ini modifications have been made
- Modify the swss\_ini.ini file
  - Find the entry "Case Program Code"
  - Put "CFC" as the value for "Case Program Code"
  - Find the entry "Case Status"
  - Put 400 as the value for "Case Status"
  - Find the entry "Update Mode"
  - Put "NU" as the value for "Update Mode"
  - Update the global variables that hold the value of "Update Mode" and "Case Status"

- If this is a CFC case and legal data is being shared with companion cases, make the same changes for each companion case.

**If the termination petition is denied (order type 29 is entered):**

- Update the database
  - Set the end\_date in load\_case for the secondary worker to the date of the order
  - Update swss\_case
    - Set the swss\_case\_state to reflect the case is no longer referred/accepted/assigned by the Adoption unit.
- Notify the Adoption supervisor and worker of the denial; this should be done via email.
- If this is a CFC case and legal data is being shared with companion cases, make the same changes for each companion case.

**If the adoption is finalized (order type 20 is entered):**

- Update the database:
  - Update swss\_case
    - Set swss\_case\_state to 600 (closed)
    - Set close\_date\_entered to system date
    - Set close\_code to '01'
    - Set program\_status to 1
  - Update provider\_case\_person
    - Set prov\_end\_date to the date of the order
    - Set prov\_end\_date\_entered to system date
    - Set adoption\_end\_reason to 'F'
- Modify the swss\_ini.ini file
  - Find the entry "Close\_Date"
  - Set the value for "Close\_Date" to the date of the order
  - Find the entry "Close\_Code"
  - Set the value for "Close\_Code" to "1"
  - Find the entry "Miscellaneous"
  - Set the value for "Miscellaneous" to "Close Case"
  - Find the entry "Source Process"
  - Set the value for "Source Process" to legal.exe
- Call the process to generate the closing FIA5S
- Modify the swss\_ini.ini file
  - Find the entry "Update Mode"
  - Set the value for "Update Mode" to "NU"
- Remind the worker, using the msgbox, to do a closing summary

**If the case is referred to Adoption:**

- Update swss\_case:
  - Set swss\_case\_state to 475 if the current swss\_case\_state is 300
  - Set swss\_case\_state to 450 if the current swss\_case\_state is 200
  - Set latest\_legal\_status to the value entered in frmFCLegal.cboLegStat1.
- Create a new ad\_info column; insert the following columns:
  - INSERT INTO ad\_info
    - (county\_no,
    - ad\_info\_id,
    - log\_id,

- refer\_to\_adopt\_date,  
• mare\_status)
- Delete the tickler for referring to adoption:
  - DELETE FROM worker\_std\_tickler
  - WHERE county\_no = i\_county\_no
  - AND log\_id = i\_log\_id
  - AND UPPER(std\_tickler\_id) = 'REFTOADOP';

**If the case is placed for Adoption (order type 17, 18, or 19 entered):**

- Create a row in ad\_info for this case
- In swss\_case table:
  - Change the program\_code to "ADPT"
  - Change the swss\_case\_state to 300 (registered status)
  - Change the primary\_worker\_id to the new primary worker
- Find the current placement (prov\_end\_date will be null) in provider\_case\_person and update the prov\_end\_date and prov\_end\_date\_entered with the date of the order.
- In load\_case table:
  - Find the secondary worker and if one is found:
    - Find the current primary worker and update the end\_date with the order date.
    - Update the secondary worker's end\_date with the order date
    - Insert a record in load\_case making the secondary worker the primary worker
- Save some of the child's CFC info from swss\_case and group\_person in swss\_conversion and aka\_person\_name (in case the adoption is disrupted):
  - From swss\_case
    - cis\_case\_no
  - From group\_person
    - recipient\_id
    - last\_name
    - last\_name\_soundex,
    - first\_name,
    - first\_name\_soundex,
    - middle\_name,
    - name\_suffix,
    - swss\_case.sib\_grp\_id,
    - ssn
  - Save in swss\_conversion
    - Cis\_cfc\_case\_no
    - Cfc\_recipient\_id
  - Save in aka\_person\_name
    - sib\_grp\_id,
    - aka\_person\_no,
    - person\_id,
    - first\_name,
    - middle\_name,
    - last\_name,
    - name\_suffix,
    - first\_name\_soundex,
    - last\_name\_soundex,
    - prior\_to\_adop\_ind,
    - old\_ssn

- Update group\_person table
  - Set other\_names to 'Y'
  - Set cfc recip id to Null
  - Set ssn to Null
  - If new name was entered, update the name fields
- Update the child's address with the address for the FIA County office
- If MA is active:
  - In swss\_case, set ma\_case\_state to 600
  - In medicaid\_history
    - Set end\_date to the date of the order
    - Set neg\_action\_date to the date of the order
    - Set neg\_action\_code to '89'
  - If there is other insurance
    - In health\_coverage, set cvg\_end\_date to the date of the order (do for both primary and secondary insurance coverages)
- If MA is registered
  - Set ma\_case\_state to 800
  - Set ma\_denial\_code to 507
  - Set ma\_withdrawal\_date to the date of the order
  - Check for open records in medicaid\_history and health\_coverage; if any are found update as specified above for Active MA.
- UPDATE cfc\_info table
  - Change goal\_status\_code to 1
  - Change goal\_code to 13
- Modify the swss\_ini file before calling the FIA5S executable:
  - Add an entry to the swss\_ini.ini file labeled "Close\_Date"
  - Put the value in Frm.mskOrdDate as the value for "Close\_Date"
  - Add an entry to the swss\_ini.ini file labeled "Close\_Code"
  - Put the value "3" as the value for "Close\_Code"
  - Add an entry to the swss\_ini.ini file labeled "Miscellaneous"
  - Put the value "Close Case" as the value for "Miscellaneous"
  - Add an entry to the swss\_ini.ini file labeled "Goal Status Code"
  - Put the value "3" as the value for "Goal Status Code"
  - Find the section labeled "OTHER" in the swss\_ini.ini file
  - Put calling program in misc
- Call the executable to print the FIA5S to close the Foster Care case on CIS
- Call the executable to print the FIA133 to register the adoption case on ASSIST
- Modify the swss\_ini.ini file:
  - Find the entry "Case Program Code"
  - Set the program code to "ADPT"
  - Find the entry "Secondary Worker 1"
  - Set the value to ""
  - If the user signed in is not the secondary worker:
    - Find the entry "Update Mode"
    - Set the value to "NU"
  - If the user signed in is the secondary worker:
    - Find the entry "Update Mode"
    - Set the value to "U"
- Re-read the swss\_ini.ini file so the global variables used are reset to the new values.  
If this is a CFC case and legal data is being shared with companion cases, make the same changes for each companion case.

**If jj case and there are victims, update jj\_info**

- UPDATE jj\_info
  - SET notify\_victims to "Y" if there are victims who want to be notified of changes in the youth's circumstances.
  - SET notify\_victims to "N" if there are no victims to be notified of changes in the youth's circumstances.

**If dual legal status and CFC worker enters order type 15 (discharge):**

- The CFC worker is removed as secondary worker
  - UPDATE load\_case
    - SET end\_date to the value of the order date for the worker with responsibility "S" and end\_date is NULL

**If the child's legal status is changing:**

- Update swss\_case
  - Set latest\_legal\_status to the value of the cboLegStat1 list box.

**Business Events**

The following are business events that occur which will initiate use of the Legal section in SWSS Childrens :

- Legal representation data needs adding/updating
- New petitions are filed with the court system
- New hearings are held with the court system
- A new AFCARS episode has begun and removal conditions need to completed

1.3.3. **List of Program Units** – If you’re smart, you’ll do a “Find” in VB to make sure the call you’re looking at isn’t in any other sections of code other than the ones listed below.

FORM/MODULE	FUNCTION	SUB FUNCTION	DATABASE CALLS
frmlegal.frm	cmdRemove.click	retrieve_removals get_members legal.bas/get_companions get_removal_cond	packlists.get_fam_struct
	CmdRepresent.click	Get_reps get_representation_data clear_rep_screen pop_representative_type pop_state_box	packlists.get_state packlists.get_members
	CmdContinue.click	Get_representation_data Insert_legal_rep Get_reps Return_from_rep Edit_rep_data Insert_conditions Leaving_legal	
	Form.load	Leaving_legal Get_legal_history Get_misc_legal Check_display Legal.bas/check_for_removal_conditions Swss.bas/SetDeviceIndependentWindow	
	Form.activate	Get_misc_legal Get_legal_history Check_display Swss.bas/ExtractINI_info	
	CmdNewPetition.click	Clear_fields	
	CmdCancel.click	Return_from_rep Leaving_legal	
	CmdUpdate.click	Clear_rep_screen	
	GrdRepSummary.click	Clear_rep_screen	
FrmFCLegal	CmdContinue.click	Update_comp_legal Display_next_panel Edit_fc_data Check_share_data Legal.bas/maintain_legal Update_comp_legal Legal.bas/print_next_hearing Legal.bas/print_pymt_notice	
	CboLegStat1.click	Hide_fields_for_41	

	CboLegStat1.LostFocus	Hide_fields_for_41	
	CboOrdType.click	Check_fc_order_type	
	CmdComments.click	Legal.bas/maintain_comments	
	Form.activate	Swss.bas/ExtractINI_Info Swss.bas/SetDeviceIndependentWindow	
	Form.load	Fill_list_boxes Legal.bas/check_refer_question Legal.bas/move_array_data Legal.bas/get_specific_record Legal.bas/fill_screen Display_court Swss.bas/EnableControlsOn Legal.bas/get_companions	
	CboOrdType.lost_focus	Check_fc_order_type	
FrmJJLegal	ChkViewOffenses.click		legal.get_all_offense_codes
	CmdContinue.click	Edit_jj_data Legal.bas/maintain_legal Legal.bas/print_next_hearing Legal.bas/print_pymt_notice Display_next_panel	
	CmdUpdVictim.click	Insert_jj_victim Edit_victim_data	
	CmdAddVictim.click	Insert_jj_victim Edit_victim_data	
	CmdComments.click	Legal.bas/maintain_comments	
	Form.activate	Swss.bas/ExtractINI_Info	
	Form.load	Fill_jj_list_boxes Swss.bas/SetDeviceIndependentWindow Legal.bas/fill_list_boxes Get_victims Move_array_data Legal.bas/get_specific_record Legal.bas/fill_screen Display_del_court Swss.bas/EnableControlsOn	
Legal.bas	Insert_jj_commitment		legal.insert_commitment
	Delete_commitment		legal.insert_commitment
	Insert_commitment		legal.insert_commitment
	Insert_court_petition		legal.insert_court_petition
	Insert_hearing		legal.insert_hearing
	Fill_list_boxes	Pop_order_type Pop_petition_type Pop_legal_box Pop_commitment_type	Packlists.get_courts Packlists.get_court_order_types Packlists.get_petition_types Packlists.get_legal_status Packlists.get_commitment_types Packlists.get_court_types Packlists.get_county Packlists.get_state

		Pop_court_indicator Pop_county_box Pop_state_box	
	Maintain_ticklers	Insert_tickler Update_tickler	
	Insert_tickler	swss.bas/bPutTWorkerStdTickler swss.bas/PrivateINI_Register swss.bas/tPrivateINI_GetString	
	Check_for_tickler		Placement.check_for_tickler
	Update_tickler	swss.bas/bUpdateWorkerStdTickler swss.bas/PrivateINI_Register swss.bas/tPrivateINI_GetString	
	Delete_tickler	swss.bas/bDeleteWorkerStdTickler swss.bas/PrivateINI_Register swss.bas/tPrivateINI_GetString	
	Maintain_legal	Insert_misc Insert_petition Insert_hearing Insert_order Insert_order_type Check_jj_commits Check_fc_commits Insert_court_petition Swss.bas/PrivateINI_Register Swss.bas/tPrivateINI_PutString Commonshell ExtractININ_Info Maintain_ticklers	
	Print_next_hearing	Print.bas/retrieve_standard_print Print.bas/format_worker_info_for_print Print.bas/retrieve_current_place	ment
	Print_pymt_notice	Print.bas/retrieve_standard_print Print.bas/format_worker_info_for_print Print.bas/retrieve_current_place	ment
	Maintain_comments	Insert_comment Swss.bas/PrivateINI_Register Swss.bas/tPrivateINI_putString LaunchEXE	
	Insert_comment		Longpkg.insert_rec
	Check_legal_vs_age		Print5s.ls_age_valid
	Check_fc_commits	Insert_commitment	
	Check_subsequentOffense	Update_sub_offense	
	Update_sub_offense		Legal.insert_offense
	Check_jj_commits	Check_subsequentOffense	

		Insert_jj_commitment Delete_commitment Insert_jj_offenses Delete_jj_offenses	
	Delete_jj_offenses		Legal.insert_offense
	Insert_jj_offenses		Legal.insert_offense
	Insert_jj_victim	Get_victims	Legal.insert_victim
	Insert_petition		Legal.insert_petition
	Insert_order_type		Legal.insert_order_type
	Insert_misc	PrivateINI_Register LPrivateINI_PutString CommonShell PrivateINI_DeleteEntry	Legal.update_misc_legal_info
	Insert_order		Legal.insert_order
	Fill_screen	Fill_jj_commits Fill_jj_petition Fill_fc_commits Check_conversion_prefills EnableControlsOn	
	Check_for_removal_conditions		Legal.check_for_removal_cond
	Get_removal_cond		Legal.check_for_removal_cond
	Get_supervisors		Packlists.get_supervisors
	Get_commitments		legal.get_commitments
	Get_companions		Legal.get_legal_companions
	Get_specific_record		legal.get_specific_legal_rec legal.get_specific_orders legal.get_offenses
	Get_victims		legal.get_victims
	Leaving_legal	swss.bas/AutoFlow	
	Main	swss.bas/ExtractINI_Info swss.bas/makeconnection	
	Pop_hearing_outcomes		packlists.get_hearing_outcomes
	Pop_offenses		packlists.get_offense_types
	PopRepresentative_type		packlists.get_legal_rep_types

#### **1.3.4. Report (output) Images**

Print out versions of each output report generated by the module. For each image, explain its usage.

#### **1.3.5. Data Elements**

For every element on a screen output report:

- Map each data element displayed, printed, or entered to the database table and field, such as the example done by SDM that Paula mentioned in the team meeting (p:\users\share\servwork\SWSS\templates\ReqTemplates\DataDefinitions.xls)
- Make a specific reference to the SWSS Schema data dictionary, which ought to be available any day now.
- List and discuss any specific validation routines, constraints, or dependent data validations (like legal status and living arrangement) that are not in the data dictionary. You can check (and copy from) the User Requirements Data Element Description for these type of validations.
- Specify which elements can only be changed in “Corrections” mode.
- Explain the instancing of this data element in laymen’s terms. This is implied in the table name, usually, such as the “Case\_Person” table refers to an instance of a (group) person record in a particular case, and the “Group\_Person” table refers to an instance of a person in a particular sibling group. Go ahead and say it like that, as it applies. This includes “historical” data, such as Medicaid\_History, which is an instance of medicaid data over time.

#### **1.3.6. Integration with Existing System**

How does this module integrate with SWSS. Is it a selection from the main menu? Are there specific things needed in the SWSS\_INI.ini? Are there short cut keys or menu selections? Can it be called directly from another module, during which time the calling module shows a blank screen or mess with the task bar, and does it need to update something for that calling module?

#### **1.3.7. Module Dependencies**

If this module depends on other modules, or if other modules depend on this module, to an extent beyond what has been described in the “Integration with Existing System” section above, such as Medicaid requiring both Placement and Funding data, list and explain those dependencies here.

#### **1.3.8. Database Subject Area**

- Show the tables with fields read, created, or modified here. (CRUD matrix, if you want to be fancy.)
- List the STORED PROCEDURES with detailed descriptions of how that stored procedure works (we’re serious. We really need it!)
- Use the Interface specifications that were generated (by Ed?) for the stored procedure calls that the VB module makes.

```
PACKAGE    LEGAL
AS
--*****
-- Get legal representatives
--*****

TYPE Rep_Rec IS RECORD
  (o_rep_id      legal_rep.rep_id%TYPE,
   o_last_name   legal_rep.last_name%TYPE,
   o_first_name  legal_rep.first_name%TYPE,
   o_middle_name legal_rep.middle_name%TYPE,
   o_phone_id    legal_rep_phone.phone_id%TYPE,
   o_phone_number phone.phone_no%TYPE,
   o_address_id  legal_rep_address.address_id%TYPE,
   o_line1       address.line1%TYPE,
   o_line2       address.line2%TYPE,
   o_city        address.city%TYPE,
   o_state_code  address.state_code%TYPE,
   o_zip         address.zip%TYPE,
   o_zip_plus    address.zip_plus%TYPE,
   o_country     address.country%TYPE,
   o_province    address.province%TYPE,
   o_alt_phone_id legal_rep_phone.phone_id%TYPE,
   o_alt_phone_no VARCHAR2(14),
   o_fax_id      legal_rep_phone.phone_id%TYPE,
   o_fax_no      VARCHAR2(30),
   o_foreign_phone_id legal_rep_phone.phone_id%TYPE,
   o_foreign_phone  phone.foreign_phone%TYPE,
   o_foreign_alt_id legal_rep_phone.phone_id%TYPE,
   o_foreign_alt_phone phone.foreign_phone%TYPE);
--  o_foreign_fax  VARCHAR2(30);
```

```
TYPE Reps_Cursor IS REF CURSOR RETURN Rep_Rec;
```

```
PROCEDURE GET_REPS
  (GetRepsCursor IN OUT LEGAL.Reps_Cursor);
```

```
--*****
--  GET COMPANION LOG_IDs AND NAMES
--*****
```

```
TYPE Legal_Comp_Rec IS RECORD
  (o_sib_person_id  swss_case.child_id%TYPE,
   o_sib_log_id    swss_case.log_id%TYPE,
   o_sib_county_no swss_case.log_id%TYPE,
   o_sib_name      VARCHAR2(50),
   o_sib_case_state swss_case.swss_case_state%TYPE,
   o_sib_prg_code  swss_case.program_code%TYPE,
   o_sib_second_wkr VARCHAR2(10));
```

```
TYPE Legal_Comp_Cursor IS REF CURSOR RETURN Legal_Comp_Rec;
```

```
PROCEDURE GET_LEGAL_COMPANIONS
  (i_person_id      IN swss_case.child_id%TYPE,
   i_log_id        IN swss_case.log_id%TYPE,
   i_county_no     IN swss_case.county_no%TYPE,
   GetLegalCompCursor IN OUT LEGAL.Legal_Comp_Cursor);
```

```
--*****
-- Get removal conditions
--*****
TYPE Removal_Cond_Rec IS RECORD
  (o_removal_reason_cd  removal_reasons.removal_reason_cd%TYPE,
   o_removal_reason_text  removal_reason_codes.removal_reason_text%TYPE);

TYPE Removal_Conds_Cursor IS REF CURSOR RETURN Removal_Cond_Rec;

PROCEDURE GET_REMOVAL_COND
  (i_placement_id      IN provider_case_person.placement_id%TYPE,
   o_removal_seq_no    OUT removal.removal_seq_no%TYPE,
   o_family_structure_code OUT removal.family_structure_code%TYPE,
   o_primary_caretaker  OUT VARCHAR2,
   o_secondary_caretaker OUT VARCHAR2,
   o_entered_date      OUT VARCHAR2,
   GetRemovalCondsCursor IN OUT LEGAL.Removal_Conds_Cursor);

--*****
-- Get legal history for a child
--*****
TYPE Legal_History_Rec IS RECORD
  (o_county_no        petition_case_person.county_no%TYPE,
   o_petition_id       petition_case_person.petition_id%TYPE,
   o_log_id            petition_case_person.log_id%TYPE,
   o_petition_type_code petition.petition_type_code%TYPE,
   o_petition_text     petition_type.description%TYPE,
   o_petition_date     VARCHAR2(10),
   o_hearing_no        hearing.hearing_no%TYPE,
   o_hearing_date      VARCHAR2(10),
   o_next_hearing_date VARCHAR2(20),
   o_next_hearing_time VARCHAR2(8),
   o_court_order_no    court_order.court_order_no%TYPE,
   o_legal_status_cd   order_case_person.legal_status_cd%TYPE,
   o_acceptance_date   VARCHAR2(10),
   o_conversion_origin_ind order_case_person.conversion_origin_ind%TYPE,
   o_judicial_determ   order_case_person.judicial_determ%TYPE,
   o_reasonable_effort order_case_person.reasonable_effort%TYPE,
   o_efrts_to_rtn_home order_case_person.efrts_to_rtn_home%TYPE);

TYPE Legal_History_Cursor IS REF CURSOR RETURN Legal_History_Rec;

PROCEDURE GET_LEGAL_HISTORY
  (i_person_id        IN petition_case_person.person_id%TYPE,
   i_log_id           IN petition_case_person.log_id%TYPE,
   i_county_no        IN petition_case_person.county_no%TYPE,
   o_dob              OUT VARCHAR2,
   o_elig_code        OUT swss_case.eligibility_code%TYPE,
   o_living_arrangement OUT swss_case.latest_living_arrangement%TYPE,
   o_num_petitions    OUT NUMBER,
   o_num_hearings     OUT NUMBER,
   o_num_orders       OUT NUMBER,
   o_accept_date      OUT VARCHAR2,
   o_mom_terminated   OUT VARCHAR2,
   o_dad_terminated   OUT VARCHAR2,
```

```
o_restitution_req      OUT VARCHAR2,
o_paying_rest         OUT VARCHAR2,
LegalHistoryCursor   IN OUT LEGAL.Legal_History_Cursor);

--*****
-- Get order types to go with the legal history records for a child
--*****

TYPE Order_Types_Rec IS RECORD
  (o_match_order_no      court_order.court_order_no%TYPE,
   o_order_type_code     court_ord_ord_type.order_type_code%TYPE,
   o_order_type_text     court_order_type.order_type_text%TYPE);

TYPE Order_Types_Cursor IS REF CURSOR RETURN Order_Types_Rec;

PROCEDURE Get_All_Order_Types
  (i_person_id          IN petition_case_person.person_id%TYPE,
   i_log_id              IN petition_case_person.log_id%TYPE,
   i_county_no           IN petition_case_person.county_no%TYPE,
   OrderTypesCursor      IN OUT Order_Types_Cursor);

--*****
-- GET A SPECIFIC COURT RECORD
--*****

TYPE Specific_Legal_Rec IS RECORD
  (court_no              court.court_no%TYPE,
   docket_no              court.petition.docket_no%TYPE,
   court_type_code        court.petition.court_type_code%TYPE,
   court_name              court.court_name%TYPE);

TYPE Specific_Legal_Cursor IS REF CURSOR RETURN Specific_Legal_Rec;

PROCEDURE GET_SPECIFIC_LEGAL_REC
  (i_county_no            IN petition.county_no%TYPE,
   i_petition_id          IN petition.petition_id%TYPE,
   i_hearing_no            IN hearing.hearing_no%TYPE,
   i_court_order_no        IN court_order.court_order_no%TYPE,
   i_log_id                IN order_case_person.log_id%TYPE,
   i_person_id              IN order_case_person.person_id%TYPE,
   o_acceptance_date       OUT VARCHAR2,
   o_legal_status_cd       OUT order_case_person.legal_status_cd%TYPE,
   o_mandated_petition     OUT petition.mandated_petition%TYPE,
   o_county_of_referral    OUT petition.county_of_referral%TYPE,
   o_report_due_date       OUT VARCHAR2,
   o_type_of_hearing        OUT hearing.type_of_hearing%TYPE,
   o_hearing_outcomes      OUT hearing_case_person.hearing_outcome_code%TYPE,
   o_order_date              OUT VARCHAR2,
   o_reasonable_effort      OUT order_case_person.reasonable_effort%TYPE,
   o_judicial_determ        OUT order_case_person.judicial_determ%TYPE,
   o_ct_specified_plcmnt    OUT order_case_person.ct_specified_plcmnt%TYPE,
   o_restitution_required    OUT order_case_person.restitution_required%TYPE,
   o_paying_restitution      OUT order_case_person.paying_restitution%TYPE,
   o_rest_paid_full_date    OUT VARCHAR2,
   o_order_disposition_dt    OUT VARCHAR2,
   o_adoption_referral       OUT order_case_person.adoption_referral%TYPE,
   o_efrts_to_rtn_home       OUT order_case_person.efrts_to_rtn_home%TYPE,
   o_m_rights_term_date      OUT VARCHAR2,
```

```
o_f_rights_term_date      OUT  VARCHAR2,
o_rights_appealed        OUT  order_case_person.rights_appealed%TYPE,
o_appeal_resolved_dt     OUT  VARCHAR2,
o_conversion_origin_ind  OUT  order_case_person.conversion_origin_ind%TYPE,
o_comment_id              OUT  hearing.comment_id%TYPE,
o_court_order_row         OUT  order_case_person.row_number%TYPE,
o_shared_already          OUT  VARCHAR2,
o_link_ind                OUT  case_link.link_ind%TYPE,
SpecificLegalCursor       IN OUT Specific_Legal_Cursor);

-- *****
-- GET Order types for a SPECIFIC COURT RECORD
-- *****

TYPE Specific_Court_Orders_Rec IS RECORD
(o_order_type_code    court_ord_ord_type.order_type_code%TYPE);

TYPE Specific_Court_Orders_Cursor IS REF CURSOR RETURN
Specific_Court_Orders_Rec;

PROCEDURE Get_Specific_Orders
(i_county_no           IN   court_ord_ord_type.county_no%TYPE,
i_court_order_no       IN   court_ord_ord_type.court_order_no%TYPE,
SpecificCourtOrdersCursor IN OUT Specific_Court_Orders_Cursor);

-- *****
-- Get all offense codes for a youth
-- *****

TYPE Offense_Code_Rec IS RECORD
(o_offense_date        VARCHAR2(10),
o_offense_type_code   offense.offense_type_code%TYPE,
o_arrest_date          VARCHAR2(10),
o_description          VARCHAR2(200),
o_sex_offense_register offense.sex_offense_register%TYPE,
o_court_order_no       court_order.court_order_no%TYPE,
o_log_id               petition_case_person.log_id%TYPE);

TYPE All_Offense_Codes_Cursor IS REF CURSOR RETURN Offense_Code_Rec;

PROCEDURE GET_ALL_OFFENSE_CODES
(i_person_id            IN   offense.person_id%TYPE,
GetAllOffenseCodesCursor IN OUT LEGAL.All_Offense_Codes_Cursor);

-- *****
-- For a log, retrieve all records of legal representatives
-- *****

TYPE Rep_Whom_Rec IS RECORD
(o_leg_rep_seq_no      represent_whom.leg_rep_seq_no%TYPE,
o_rep_id               legal_rep.rep_id%TYPE,
o_rep_name              VARCHAR2(40),
o_begin_date            VARCHAR2(10),
o_end_date              VARCHAR2(10),
o_rep_whom              represent_whom.rep_whom%TYPE,
o_legal_rep_code        represent_whom.legal_rep_code%TYPE);

TYPE Rep_Whom_Cursor IS REF CURSOR RETURN Rep_Whom_Rec;
```

```
PROCEDURE GET_REP_WHOM
    (i_county_no    IN represent_whom.county_no%TYPE,
     i_log_id      IN represent_whom.log_id%TYPE,
     i_person_id   IN represent_whom.person_id%TYPE,
     GetRepWhomCursor IN OUT LEGAL.Rep_Whom_Cursor);

--*****
-- For a log, retrieve all victim records
--*****

TYPE Victim_Rec IS RECORD
    (o_victim_no        jj_victim.victim_no%TYPE,
     o_victim_last_name jj_victim.victim_last_name%TYPE,
     o_victim_first_name jj_victim.victim_first_name%TYPE,
     o_address_id       legal_rep_address.address_id%TYPE,
     o_line1            address.line1%TYPE,
     o_line2            address.line2%TYPE,
     o_city              address.city%TYPE,
     o_state_code        address.state_code%TYPE,
     o_zip                address.zip%TYPE,
     o_zip_plus          address.zip_plus%TYPE,
     o_phone_id          legal_rep_phone.phone_id%TYPE,
     o_phone_number      VARCHAR2(14),
     o_notify_victim    jj_cp_victim.notif_by_letter_phone%TYPE,
     o_country           address.country%TYPE,
     o_province          address.province%TYPE,
     o_foreign_phone     phone.foreign_phone%TYPE);

TYPE Victims_Cursor IS REF CURSOR RETURN Victim_Rec;

PROCEDURE GET_VICTIMS
    (i_county_no    IN jj_cp_victim.county_no%TYPE,
     i_log_id      IN jj_cp_victim.log_id%TYPE,
     i_person_id   IN jj_cp_victim.person_id%TYPE,
     GetVictimsCursor IN OUT LEGAL.Victims_Cursor);

--*****
-- Get all offense codes for a particular court order
--*****

TYPE Offense_Rec IS RECORD
    (o_offense_date    VARCHAR2(10),
     o_offense_type_code offense.offense_type_code%TYPE,
     o_arrest_date      VARCHAR2(10),
     o_sex_offense_register offense.sex_offense_register%TYPE);

TYPE Offenses_Cursor IS REF CURSOR RETURN Offense_Rec;

PROCEDURE GET_OFFENSES
    (i_county_no    IN offense.county_no%TYPE,
     i_log_id      IN offense.log_id%TYPE,
     i_person_id   IN offense.person_id%TYPE,
     i_court_order_no IN offense.court_order_no%TYPE,
     GetOffensesCursor IN OUT LEGAL.Offenses_Cursor);

--*****
-- Get all courts
--*****
```

```
TYPE Court_Rec IS RECORD
(o_court_no      court.court_no%TYPE,
o_court_name    court.court_name%TYPE,
o_county_no     petition_case_person.county_no%TYPE,
o_line1         address.line1%TYPE,
o_line2         address.line2%TYPE,
o_city          address.city%TYPE,
o_state_code    address.state_code%TYPE,
o_zip           address.zip%TYPE,
o_zip_plus      address.zip_plus%TYPE);

TYPE Court_Cursor IS REF CURSOR RETURN Court_Rec;

PROCEDURE GET_COURTS
(GetCourtsCursor IN OUT Court_Cursor);

-- ****
-- Retrieve all commitment codes for a court order
-- ****

TYPE Commitment_Rec IS RECORD
(o_commit_type_code  commitment.commit_type_code%TYPE,
o_commitment_date   VARCHAR2(10),
o_cmtmnt_accept_date VARCHAR2(10),
o_commitment_county commitment.commitment_county%TYPE,
o_charged_county    commitment.charged_county%TYPE,
o_commitment_row    NUMBER);

TYPE Commitment_Cursor IS REF CURSOR RETURN Commitment_Rec;

PROCEDURE GET_COMMITS
(i_county_no      IN commitment.county_no%TYPE,
i_log_id         IN commitment.log_id%TYPE,
i_person_id      IN commitment.person_id%TYPE,
i_legal_status_cd IN commitment.legal_status_cd%TYPE,
i_court_order_no IN commitment.court_order_no%TYPE,
GetCommitmentsCursor IN OUT LEGAL.Commitment_Cursor);

-- ****
-- NEW REF CURSOR SPEC FOR PROCEDURE ARE_THERE_ANY_OFFENSES
-- ****

TYPE Any_Offenses_Rec IS RECORD
(o_commit_type_code  commitment.commit_type_code%TYPE,
o_commitment_date   VARCHAR2(10),
o_cmtmnt_accept_date VARCHAR2(10),
o_commitment_county commitment.commitment_county%TYPE,
o_charged_county    commitment.charged_county%TYPE,
o_commitment_row    NUMBER,
o_legal_status_cd   order_case_person.legal_status_cd%TYPE,
o_court_order_no    court_order.court_order_no%TYPE);

TYPE Any_Offenses_Cursor IS REF CURSOR RETURN Any_Offenses_Rec;

PROCEDURE are_there_any_offenses
(i_county_no      IN commitment.county_no%TYPE,
i_log_id         IN commitment.log_id%TYPE,
i_person_id      IN commitment.person_id%TYPE,
```

```
o_offense_ind      OUT  VARCHAR2,
o_sex_offender     OUT  VARCHAR2,
o_sex_offense_register  OUT  VARCHAR2,
i_legal_status     IN   swss_case.latest_legal_status%TYPE,
AreThereAnyOffensesCursor IN OUT LEGAL.Any_Offenses_Cursor);

--*****
-- Purpose: Check if removal conditions have been completed for the
-- latest placement if it begins a new episode
--*****
PROCEDURE    CHECK_FOR_REMOVAL_COND
(i_county_no    IN removal.county_no%TYPE,
i_log_id       IN removal.log_id%TYPE,
i_person_id    IN removal.person_id%TYPE,
o_new_episode   OUT provider_case_person.new_episode_ind%TYPE,
o_removal_seq_no OUT removal.removal_seq_no%TYPE,
o_count        OUT NUMBER,
o_placement_id  OUT provider_case_person.placement_id%TYPE);

--*****
-- Retrieve the current court's name
--*****
PROCEDURE    GET_CURRENT_COURT_NAME
(i_county_no    IN petition_case_person.county_no%TYPE,
i_log_id       IN petition_case_person.log_id%TYPE,
i_person_id    IN petition_case_person.person_id%TYPE,
i_legal_status  IN order_case_person.legal_status_cd%TYPE,
o_court_name   OUT court.court_name%TYPE);

--*****
-- Purpose: Get various information needed for edits
--*****
PROCEDURE    GET_MISC_LEGAL_INFO
(i_county_no    IN jj_info.county_no%TYPE,
i_log_id       IN jj_info.log_id%TYPE,
o_goal_code    OUT jj_info.goal_code%TYPE,
o_notify_victims OUT jj_info.notify_victims%TYPE,
o_cfc_log_id   OUT swss_case.log_id%TYPE,
o_cfc_county_no OUT swss_case.county_no%TYPE,
o_cfc_legal_status OUT swss_case.latest_legal_status%TYPE,
o_cfc_case_state OUT swss_case.swss_case_state%TYPE,
o_open_pymts   OUT VARCHAR2,
i_person_id    IN swss_case.child_id%TYPE,
o_adpt_wkr_email OUT address.email_address%TYPE,
o_adpt_supv_email OUT address.email_address%TYPE,
o_adpt_wkr_load  OUT VARCHAR2,
o_adpt_wkr_name  OUT VARCHAR2);

--*****
-- Purpose: Update various legal information
--*****
PROCEDURE    UPDATE_MISC_LEGAL_INFO
(i_county_no    IN jj_info.county_no%TYPE,
i_log_id       IN jj_info.log_id%TYPE,
i_person_id    IN swss_case.child_id%TYPE,
i_notify_victims IN jj_info.notify_victims%TYPE,
```

```
i_legal_status      IN    swss_case.latest_legal_status%TYPE,  
i_chg_swss_case    IN    VARCHAR2,  
i_chg_jj_info      IN    VARCHAR2,  
i_order_date       IN    VARCHAR2,  
i_chg_adpt_status IN    VARCHAR2,  
i_last_name        IN OUT group_person.last_name%TYPE,  
i_first_name       IN OUT group_person.first_name%TYPE,  
i_middle_name      IN OUT group_person.middle_name%TYPE,  
i_worker_id        IN    swss_case.primary_worker_id%TYPE,  
i_load_no          IN    VARCHAR2,  
o_chg_status       OUT   VARCHAR2,  
o_new_case_state   OUT   VARCHAR2,  
o_ma_closed        OUT   VARCHAR2);
```

```
--*****
```

```
-- Purpose: Insert juvenile justice victims
```

```
--*****
```

```
PROCEDURE    INSERT_VICTIM  
(i_victim_no      IN OUT jj_victim.victim_no%TYPE,  
 i_victim_last_name IN jj_victim.victim_last_name%TYPE,  
 i_victim_first_name IN jj_victim.victim_first_name%TYPE,  
 i_address_id     IN OUT jj_victim.address_id%TYPE,  
 i_line1          IN    address.line1%TYPE,  
 i_line2          IN    address.line2%TYPE,  
 i_city            IN    address.city%TYPE,  
 i_state_code     IN    address.state_code%TYPE,  
 i_zip             IN    address.zip%TYPE,  
 i_zip_plus        IN    address.zip_plus%TYPE,  
 i_phone_id        IN OUT jj_victim.phone_id%TYPE,  
 i_area_code       IN    phone.area_code%TYPE,  
 i_phone_no        IN    phone.phone_no%TYPE,  
 i_notify          IN    jj_cp_victim.notif_by_letter_phone%TYPE,  
 i_county_no       IN    jj_cp_victim.county_no%TYPE,  
 i_log_id          IN    jj_cp_victim.log_id%TYPE,  
 i_person_id       IN    jj_cp_victim.person_id%TYPE,  
 i_country         IN    address.country%TYPE,  
 i_province        IN    address.province%TYPE,  
 i_foreign_phone   IN    phone.foreign_phone%TYPE);
```

```
--*****
```

```
-- Purpose: Insert removal conditions
```

```
--*****
```

```
PROCEDURE    INSERT_REMOVAL_COND  
(i_county_no      IN removal_condition.county_no%TYPE,  
 i_removal_seq_no IN removal.removal_seq_no%TYPE,  
 i_removal_reason_cd IN VARCHAR2,  
 i_log_id          IN removal_condition.log_id%TYPE,  
 i_person_id       IN removal_condition.person_id%TYPE,  
 i_family_structure_code IN removal.family_structure_code%TYPE,  
 i_primary_caretaker_id IN removal.primary_caretaker_id%TYPE,  
 i_secondary_caretaker_id IN removal.secondary_caretaker_id%TYPE,  
 i_placement_id    IN provider_case_person.placement_id%TYPE,  
 o_removal_seq_no  OUT removal.removal_seq_no%TYPE);
```

```
--*****
```

```
-- INSERT A PETITION RECORD
-- ****
PROCEDURE    INSERT_PETITION
  (i_county_no      IN petition.county_no%TYPE,
   i_petition_id    IN petition.petition_id%TYPE,
   i_petition_type_code IN petition.petition_type_code%TYPE,
   i_petition_date   IN VARCHAR2,
   i_mandated_petition IN petition.mandated_petition%TYPE,
   i_county_of_referral IN petition.county_of_referral%TYPE,
   i_log_id          IN petition_case_person.log_id%TYPE,
   i_person_id        IN petition_case_person.person_id%TYPE,
   o_petition_id     OUT petition.petition_id%TYPE);

-- ****
-- Insert court orders types
-- ****
PROCEDURE    INSERT_ORDER_TYPE
  (i_county_no      IN court_ord_ord_type.county_no%TYPE,
   i_order_type_code IN court_ord_ord_type.order_type_code%TYPE,
   i_court_order_no   IN court_ord_ord_type.court_order_no%TYPE,
   i_action          IN VARCHAR2);

-- ****
-- Purpose: insert data into court order table
-- ****
PROCEDURE    INSERT_ORDER
  (i_county_no      IN court_order.county_no%TYPE,
   i_court_order_no   IN court_order.court_order_no%TYPE,
   i_legal_status_cd  IN order_case_person.legal_status_cd%TYPE,
   i_order_date       IN VARCHAR2,
   i_petition_id      IN petition.petition_id%TYPE,
   i_hearing_no        IN court_order.hearing_no%TYPE,
   i_reasonable_effort IN order_case_person.reasonable_effort%TYPE,
   i_judicial_determ  IN order_case_person.judicial_determ%TYPE,
   i_specified_plcmnt IN order_case_person.ct_specified_plcmnt%TYPE,
   i_restitution_required IN order_case_person.restitution_required%TYPE,
   i_paying_restitution IN order_case_person.paying_restitution%TYPE,
   i_rest_paid_full_date IN VARCHAR2,
   i_order_disposition_dt IN VARCHAR2,
   i_adoption_referral  IN order_case_person.adoption_referral%TYPE,
   i_efrts_to rtn_home IN order_case_person.efrts_to rtn_home%TYPE,
   i_m_rights_term_date IN VARCHAR2,
   i_f_rights_term_date IN VARCHAR2,
   i_rights_appealed   IN order_case_person.rights_appealed%TYPE,
   i_appeal_resolved_dt IN VARCHAR2,
   i_conversion_origin_ind IN order_case_person.conversion_origin_ind%TYPE,
   i_acceptance_date   IN VARCHAR2,
   i_log_id            IN order_case_person.log_id%TYPE,
   i_person_id          IN order_case_person.person_id%TYPE,
   i_court_order_row    IN NUMBER,
   i_old_legal_status   IN VARCHAR2,
   o_court_order_no     OUT court_order.court_order_no%TYPE);

-- ****
-- Purpose: insert data into offense table
-- ****
```

```
PROCEDURE    INSERT_OFFENSE
(i_county_no      IN offense.county_no%TYPE,
i_offense_date    IN VARCHAR2,
i_offense_type_code IN offense.offense_type_code%TYPE,
i_log_id          IN offense.log_id%TYPE,
i_person_id       IN offense.person_id%TYPE,
i_court_order_no   IN offense.court_order_no%TYPE,
i_arrest_date     IN VARCHAR2,
i_sex_offense_register IN offense.sex_offense_register%TYPE,
i_action          IN VARCHAR2);

--*****
-- Purpose: Insert legal representatives
--*****

PROCEDURE    INSERT_LEGAL_REP
(i_rep_id         IN legal_rep.rep_id%TYPE,
i_legal_rep_code  IN represent_whom.legal_rep_code%TYPE,
i_last_name       IN legal_rep.last_name%TYPE,
i_first_name      IN legal_rep.first_name%TYPE,
i_middle_name     IN legal_rep.middle_name%TYPE,
i_address_id      IN address.address_id%TYPE,
i_addr_type_code  IN legal_rep_address.addr_type_code%TYPE,
i_line1           IN address.line1%TYPE,
i_line2           IN address.line2%TYPE,
i_city             IN address.city%TYPE,
i_state_code       IN address.state_code%TYPE,
i_zip              IN address.zip%TYPE,
i_zip_plus         IN address.zip_plus%TYPE,
i_phone_id         IN phone.phone_id%TYPE,
i_phone_type_code  IN legal_rep_phone.phone_type_code%TYPE,
i_area_code        IN phone.area_code%TYPE,
i_phone_no         IN phone.phone_no%TYPE,
i_extension        IN phone.extension%TYPE,
i_county_no        IN represent_whom.county_no%TYPE,
i_leg_rep_seq_no   IN represent_whom.leg_rep_seq_no%TYPE,
i_log_id           IN represent_whom.log_id%TYPE,
i_person_id        IN represent_whom.person_id%TYPE,
i_begin_date       IN VARCHAR2,
i_end_date         IN VARCHAR2,
i_rep_whom         IN represent_whom.rep_whom%TYPE,
i_alt_phone_id     IN legal_rep_phone.phone_id%TYPE,
i_alt_area_code    IN phone.area_code%TYPE,
i_alt_phone_no     IN phone.phone_no%TYPE,
i_fax_id           IN legal_rep_phone.phone_id%TYPE,
i_fax_area_code    IN phone.area_code%TYPE,
i_fax_no           IN phone.phone_no%TYPE,
i_country          IN address.country%TYPE,
i_province         IN address.province%TYPE,
i_foreign_phone    IN phone.foreign_phone%TYPE,
i_foreign_alt_phone IN phone.foreign_phone%TYPE,
i_foreign_fax      IN phone.foreign_phone%TYPE,
o_rep_id           OUT legal_rep.rep_id%TYPE,
o_leg_rep_seq_no   OUT represent_whom.leg_rep_seq_no%TYPE,
o_address_id       OUT address.address_id%TYPE,
o_phone_id         OUT phone.phone_id%TYPE,
o_alt_phone_id     OUT phone.phone_id%TYPE,
```

```
o_fax_id      OUT phone.phone_id%TYPE);

-- *****
-- Purpose: insert data into hearing table
-- *****

PROCEDURE    INSERT_HEARING
(i_county_no   IN  hearing.county_no%TYPE,
i_petition_id  IN  hearing.petition_id%TYPE,
i_comment_id   IN  hearing.comment_id%TYPE,
i_hearing_no   IN  hearing.hearing_no%TYPE,
i_hearing_date  IN  VARCHAR2,
i_next_hearing_date  IN  VARCHAR2,
i_report_due_date  IN  VARCHAR2,
i_type_of_hearing  IN  VARCHAR2,
i_hearing_outcomes  IN  hearing_case_person.hearing_outcome_code%TYPE,
i_person_id    IN  hearing_case_person.person_id%TYPE,
i_log_id       IN  hearing_case_person.log_id%TYPE,
o_hearing_no   OUT  hearing.hearing_no%TYPE);

-- *****
--     INSERT A COURT PETITION RECORD
-- *****

PROCEDURE    INSERT_COURT_PETITION
(i_county_no   IN  petition.county_no%TYPE,
i_petition_id  IN  petition.petition_id%TYPE,
i_court_no     IN  court_petition.court_no%TYPE,
i_docket_no    IN  court_petition.docket_no%TYPE,
i_action       IN  VARCHAR2,
i_old_court_no  IN  court_petition.court_no%TYPE,
i_court_type_code  IN  court_petition.court_type_code%TYPE);

-- *****
--     INSERT A COURT RECORD
-- *****

PROCEDURE    INSERT_COURT
( i_court_no    IN  court.court_no%TYPE,
i_court_name   IN  court.court_name%TYPE,
o_court_no     OUT court.court_no%TYPE);

-- *****
--     Maintain legal data for companion case(s)
-- *****

PROCEDURE    INSERT_COMPANION_LEGAL
(i_county_no   IN  petition_case_person.county_no%TYPE,
i_petition_id  IN  petition_case_person.petition_id%TYPE,
i_log_id       IN  petition_case_person.log_id%TYPE,
i_person_id    IN  petition_case_person.person_id%TYPE,
i_court_order_no  IN  order_case_person.court_order_no%TYPE,
i_legal_status_cd  IN  order_case_person.legal_status_cd%TYPE,
i_hearing_no   IN  court_order.hearing_no%TYPE,
i_reasonable_effort  IN  order_case_person.reasonable_effort%TYPE,
i_judicial_determ  IN  order_case_person.judicial_determ%TYPE,
i_specified_plcmnt  IN  order_case_person.ct_specified_plcmnt%TYPE,
i_restitution_required  IN  order_case_person.restitution_required%TYPE,
i_paying_restitution  IN  order_case_person.paying_restitution%TYPE,
i_rest_paid_full_date  IN  VARCHAR2,
```

```
i_adoption_referral    IN  order_case_person.adoption_referral%TYPE,
i_efrts_to rtn_home   IN  order_case_person.efrts_to_rtn_home%TYPE,
i_m_rights_term_date IN  VARCHAR2,
i_f_rights_term_date IN  VARCHAR2,
i_rights_appealed    IN  order_case_person.rights_appealed%TYPE,
i_appeal_resolved_dt IN  VARCHAR2,
i_conversion_origin_ind IN  order_case_person.conversion_origin_ind%TYPE,
i_acceptance_date    IN  VARCHAR2,
i_court_order_row    IN  NUMBER,
i_old_legal_status    IN  order_case_person.legal_status_cd%TYPE);

--*****
-- Maintain commitment data
--*****

PROCEDURE    INSERT_COMMITMENT
(i_county_no      IN commitment.county_no%TYPE,
 i_log_id        IN commitment.log_id%TYPE,
 i_person_id     IN commitment.person_id%TYPE,
 i_legal_status_cd IN commitment.legal_status_cd%TYPE,
 i_court_order_no IN commitment.court_order_no%TYPE,
 i_commit_type_code IN commitment.commit_type_code%TYPE,
 i_commitment_date IN VARCHAR2,
 i_cmtnmt_accept_date IN VARCHAR2,
 i_commitment_county IN commitment.commitment_county%TYPE,
 i_charged_county IN commitment.charged_county%TYPE,
 i_action        IN VARCHAR2,
 i_commitment_row IN NUMBER,
 i_old_commitment_date IN VARCHAR2);

--*****
-- Get Rep phone ids
--*****


FUNCTION GetRepPhoneId
(i_rep_id      legal_rep_phone.rep_id%TYPE,
 i_type        VARCHAR2)

RETURN legal_rep_phone.phone_id%TYPE;

PRAGMA RESTRICT_REFERENCES (GetRepPhoneId, WNDS, WNPS, RNPS);

--*****
-- Get Rep phone numbers
--*****


FUNCTION GetRepPhone
(i_rep_id      legal_rep_phone.rep_id%TYPE,
 i_type        VARCHAR2)

RETURN VARCHAR2;

PRAGMA RESTRICT_REFERENCES (GetRepPhone, WNDS, WNPS, RNPS);

--*****
-- Get Secondary Worker
--*****


FUNCTION GetSecondWkr
(i_county_no    swss_case.county_no%TYPE,
```

```
i_log_id      swss_case.log_id%TYPE,  
i_child_id    swss_case.child_id%TYPE)  
  
RETURN VARCHAR2;  
  
PRAGMA RESTRICT_REFERENCES (GetSecondWkr, WNDS, WNPS, RNPS);  
  
END LEGAL;
```

### **1.3.9. Data Warehouse**

If known, explain which items are added to the data warehouse and under what conditions they are written. Hopefully we can reference a document or set of documents supported by the data warehouse.

### **1.3.10. Technical Issues**

Discuss any tricky things that happen in the module that someone who maintains the application may not recognize at first glance. Sibling group sharing, legal status switches, or reusing person IDs.

### **1.3.11. Test Plans**

#### **TEST PLAN**

#### **LEGAL**

##### **General**

1. The primary worker, the secondary worker, and/or their supervisor may update a case. For all others, the case will be “view only”.
2. Case must be registered before Legal data may be entered.
3. If case is registered, any data may be changed. After the case is active, the data that must be changed via the correction process is legal status, commitment date, petition date, and petition type.
4. Any date fields cannot be prior to the child’s date of birth.

##### **Specific Edits and Screen Displays**

1. See edits at end of document.
2. **Required** fields are petition date, petition type, legal status, court indicator code, court of jurisdiction, hearing date, and next hearing date.
3. Court indicator code is required for legal statuses 40 –52. Except if LA is 04 and legal status is 43 or 49, then indicator is not required.
4. When legal status changes to 41 or 44 and the worker states child is to be referred to adoption, verify that in swss\_case the program code changes to “ADPT” and case state changes to 450.

5. On the legal representation screen, if a judge or referee is selected, the box to pick whom they are representing should not be displaying. After a representative has been entered, a summary panel should always display when entering this screen.
6. For Removal conditions, if the living arrangement is 01 (own home), then the worker cannot complete this screen. If it is the beginning of a new episode, this screen must be completed. If filling out this screen, there must be at least one condition checked.
7. If child is committed under ACT 196 (commitment type) or if the county charged is different from the commitment county, a question should display about printing a letter to Payment Reconciliation.
8. If new AFCARS episode, determined in Placement, then Removal Conditions will need to be completed.
9. Cannot open Juvenile Justice until Foster Care is closed.
10. If legal status is 41, 44 or 52, the question about termination of parental rights should display.
11. If there is a termination of parental rights (order type 05), the question about an appeal should be displayed. If the worker answers yes, the question "Appeal resolved" should display. If yes to this question, then date resolved should display.
12. Ticklers should be created for next hearing date and court report due date. The tickler date should be one month before the due dates and automatically deleted one day after they are due.
13. Only order date for order type of "Emergency" can be prior to petition date.
14. Hearing date must be same or later than petition date.
15. All dates must be today's date or previous dates. The exception to this is next hearing date and court report due date; they can be today or later.
16. If adding a new hearing, the petition date and type should pre-fill.

### **Foster Care Specific**

1. Valid legal statuses for Foster Care are, 41, 42, 43, 44, 45, 48, and 51.
2. If court order type is 09, and legal status is 44 or 45, then panel should display for entry of commitment data.
3. Commitment type ACT 150 is invalid.
4. If commitment type is ACT 220 or ACT 296, then legal status must be 43, 44, 45 or 52.
5. If legal status is 41, acceptance date field should be displayed.
6. If legal status changes to 44 or 45, there must be an order type code of 09 and the commitment data panel should be displayed.
7. If legal status changes to 41, 44, or 52, display the question about referring to adoption.
8. Verify that if legal status changes from 42 to either 41 or 44, the goal code changed to 10 (cfc\_info).
9. If legal status changes to 41, 44, or 52, question about reasonable efforts to return child home should display.

### **Juvenile Justice Specific**

1. Valid legal statuses for Juvenile Justice are 40, 46, 47, 50, 52, 90, 91, 92, 93, 94, 95 and 96.
2. If legal status is 46 or 52, there must be an order type code of 09. If the legal status is 52, there should be 2 order types of 09.
3. If legal status is 46 or 52, then commitment panel should be displayed. If the legal status is 52, then there can be two commitment types and dates.
4. If the child is committed under ACT 150, then committing offenses should be displayed. Committing offenses do not display if committed under ACT 220 or 296.
5. A child cannot be committed under both ACT 220 and ACT 296. A child may be committed under ACT 150 and ACT 220 or ACT 150 and ACT 296.
6. After commitment data has been entered, when adding new, these will be display only. The fields to be completed will be subsequent offense and date, and re-arrest date.
7. The acceptance date in the commitment panel cannot be prior to the commitment date.
8. There must be at least one commitment offense if committed under ACT 150.
9. Re-arrest date and subsequent offense date cannot be prior to commitment date.
10. If the worker checks the victim's rights box, then panel for entry of victim information should display.
11. If committing offense is of sex offense type (105, 204, 205, 208, 311, 333, 334, or 335), then a message about youth registering with police should display.
12. If youth is required to make restitution, then a question should display asking if restitution is being paid. If not being paid, then a reminder should display, that the worker needs to file a court report.
13. A tickler needs to be generated every six months for review of restitution payment.
14. If petition type is 14 (Request for Writ), then living arrangement must be 20 (AWOL).
15. If youth is under age 12, cannot be committed under ACT 150. Youth's older than 18 may be committed to the Agency under ACT 150.

### **Old Legal Status vs New Legal Status**

1. If old status is 40, new status can be 46 or 52.
2. If old status is 41, new status can be 42, 44, 45, or 48.
3. If old status is 42, new status can be 41, 44, 45, or 48
4. If old status is 43, 44, 49, or 52, can't change.
5. If old status is 45, new status can be 42 or 44.
6. If old status is 46, new status can be 52.
7. If old status is 47, new status can be 40 or 46.
8. If old status is 48, new status can be 41, 42, 44, or 45.
9. If old status is 50, new status can be 40 or 46.

10. If old status is 51, new status can be 41, 42, 44, 45, or 48.